Fixed Points in Spectral Complexity

Sean W. Smith Carl Sturtivant

October 1992
CMU-CS-92-190

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

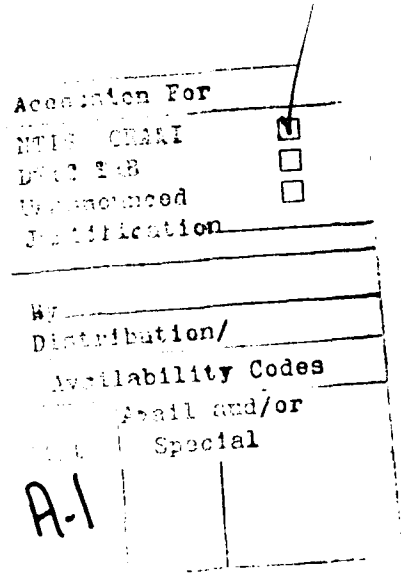
This research was sponsored by the Avionics Laboratory, Wright Research and Development Center, Aeronautical Systems Division (AFSC), U. S. Air Force, Wright-Patterson AFB, OH 45433-6543 under Contract F33615-90-C-1465, Arpa Order No. 7597. S. Smith also received support from an ONR Graduate Fellowship and from NSF Grant CCR-8858087. C. Sturtivant is currently affiliated with the Department of Computer Science at the University of Minnesota, Minneapolis MN 55455

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

Abstract

Spectral analysis has emerged as an exciting tool in complexity research. In some cases, the functions in a complexity class can be characterized by specifying a simple property of their Fourier spectra—even though their truth tables do not display such simple properties.

In this paper, we demonstrate a fundamental limitation of this tool: if a class of functions contains parity and is closed under some simple composition rules, then we can take the set of Fourier spectra of a subset of that class, close it under some simple projections, and obtain everything in the original class. Indeed, we can demonstrate this property for a general family of transforms, of which the parity-based Fourier transform is only one. If a class contains the basis function of such a transform, then no reduction in complexity is obtained when we shift from truth tables to spectra.



1. Introduction

1.1. Spectral Approaches to Complexity

Complexity classes are usually defined in terms of resources; they are the classes of functions that can be computed by some family of devices: contant-depth circuits, Turing machines with polynomial space and time, et cetera. The central problem in computational complexity is that in general it is difficult to actually characterize these classes of functions. What *really* can be computed by some given family?

Spectral analysis has recently been shown to be a way to solve this problem in some cases. Traditionally, functions are regarded as truth tables: lists of input-output pairs. There may be no apparent pattern to the set of functions calculated by some family of devices when we write the functions as truth tables. But if we look at their Fourier spectra, patterns sometimes emerge. The most well-known spectral result is Linial, Mansour and Nisan's [14] theorem that the Fourier spectrum of a function computed by an $AC0$ circuit diminishes exponentially with "frequency", but there has been a good deal of other spectral work as well [4] [6] [7] [8] [11].

What exactly is the Fourier transform of a Boolean function? If we embed the Booleans in the reals by identifying *true* and *false* with ± 1, then the real functions on n Boolean variables can be thought of as a real 2^n dimensional vector space. A natural basis is

$$\{\delta_S \ : \ S \subseteq \{1,...,n\}\}$$

where δ_S takes the characteristic vector for S to 1 and everything else to 0. This space is isomorphic to $\mathbf{R}^{2^n}$; the δ_S correspond to the standard axes, and the Boolean functions correspond to the corners of the unit hypercube. However, there is a class of parity functions—diagonals on this cube—that also forms a nice basis for this space. Researchers use the term "Fourier spectrum" to refer to the coordinates of a function with respect to this alternate basis. (Alternatively, this is known as the Walsh-Hadamard Transform [1].)

1.2. Fixed Points in Spectral Complexity

Let us first define some notation.

Definition 1 (T, TT) *If C is a collection of functions, let $TT(C)$ denote the set of truth tables of those functions. If T is a transform on the class, let $T(C)$ denote the set of spectra of functions in that class with respect to to the transform T.*

Definition 2 *If S is a set of truth tables (or spectra being regarded as truth tables), let S^* be the truth table set obtained by closing S under polynomial projection and constant depth, constant size composition.*

The main thrust behind spectral complexity research is that the set $F(C)$ of Fourier spectra of functions in some class C are somehow simpler than the set of truth tables $TT(C)$. We demonstrate a fundamental limitation of the spectral approach: under very general conditions, once a class contains parity, then a subset of its Fourier spectra looks exactly like the set of truth tables for the class. No reduction in complexity is obtained by shifting from truth tables to spectra. We also establish a similar result for a more general family of transforms.

We establish this result in the parity case by regarding Fourier spectra as truth tables and examining the eigenvectors of the Fourier transform. We obtain the surprising fact that Boolean functions can be eigenvectors—and that these functions can be arbitrarily complex. Namely, for any Boolean function f there exists a Boolean function g such that

- g is fixed under the Fourier transform—that is, its Fourier spectra differs from its truth table only by a multiplicative constant;
- f can be obtained from g by a polynomial projection;
- and computing g reduces to computing f, parity, and some two-bit functions.

Let C be a class of functions closed under constant-depth finite-sized composition and containing the two-bit functions. (Any standard complexity class has this property.) If C contains parity, then it contains the fixed point g for every f in C. Hence, the set of Fourier spectra of functions in C contains a subset of things that look just like Boolean functions. So we cannot obtain any nice dropoff invariants for the spectra. If we close this subset under polynomial projection, then we obtain exactly the set of truth tables of functions in C. That is:

$$TT(C) \subseteq F(C)^*$$

Therefore the set of spectra of functions in the class C contains a subset that is just as hard to characterize as is the set of truth tables of functions in C.

It is interesting to note that this phenomenon of spectra coinciding with truth tables occurs with the parity-based Fourier transform when the class C contains parity. Indeed, we can show that this generalizes to other transforms F_ϕ, when the class includes the basis functions ϕ of the transform. For any class C of functions.

$$\phi \in C \ \wedge C = C^* \quad \Rightarrow \quad TT(C) \subseteq F_\phi(C)^*$$

We no longer obtain individual functions as fixed points, but the notion of spectra of a class having the same complexity as the truth tables still hold.

In Section 2, we present an eigenvector-based proof of the result for the parity based Fourier transform. In Section 3, we present the general proof, which uses only the orthogonality of the transform and its self-similarity for different domain sizes. The parity theorem follows from the general theorem as a simple corollary.

2. Boolean Fixed Points of the Fourier Transform

If we identify *true* and *false* with $+1$ and -1 respectively, we can embed Boolean functions on n bits in the space of real functions on n bits. A natural way to represent a real function f on n Boolean variables is as its generalized truth table: a list of the values f takes on for the 2^n possible input vectors. Fourier analysis provides an alternative way: each subset S of $\{1,...,n\}$ defines a parity function ϕ_S, and a function f can be uniquely represented as the vector of its correlations with these functions.

If we identify subsets of $\{1,...,n\}$ with their characteristic Boolean vectors, then the Fourier transform of a function on n bits is itself a function on n bits. Regarding the Fourier transform as a map from functions to functions rather than from coordinate axes to coordinate axes provides new insight into the action of the Fourier transform. The Fourier transform F induces two orthogonal 2^{n-1} dimensional spaces of eigenvectors. Everything in the one space is fixed by F; everything in the other is negated. Hence any function may be written as the sum of a fixed and an alternating component; the transform simply negates the alternating component and multiplies everything by a positive constant.

Our research also yields the surprising result that these fixed point subspaces contain Boolean functions. In particular, every Boolean function f is the polynomial projection of some Boolean fixed point g with the property that computing g reduces to computing f and parity.

2.1. Fourier Analysis of Boolean Functions

The idea of applying Fourier analysis to Boolean functions is discussed in great detail elsewhere [14], so we shall only provide a quick overview.

First, we define the space of functions acted on by the Fourier transform.

Definition 3 *Define $\mathcal{F}(B^n, \mathbf{R})$ to be the space of real functions on n Boolean variables.*

Let us identify *true* and *false* with 1 and -1, respectively. Then $\mathcal{F}(B^n, \mathbf{R})$ also contains the Boolean functions on n bits.

Definition 4 *For $S \subseteq \{1,...,n\}$, let a_S be the characteristic Boolean vector for S. Let A_S be the function that takes a_S to 1 and any other vector to zero. Define $\phi_S(a_T) = -1^{|S \cap T|}$. For two functions $f, g \in \mathcal{F}(B^n, \mathbf{R})$ define their inner product $< f, g > = \sum_S f(a_S)g(a_S)$.*

Note that this definition also differs from that used elsewhere [14] in that we don't divide by 2^n when computing the inner product. This conveniently ensures that the inner product of Boolean functions is always an integer.

The natural representation for a function $f \in \mathcal{F}(B^n, \mathbf{R})$ is its truth table. We can restate this in terms of linear algebra: the functions A_S form an orthonormal basis for $\mathcal{F}(B^n, \mathbf{R})$ under the inner product $<>$. The main idea of Fourier analysis on Boolean functions is that the functions ϕ_S form a basis for $\mathcal{F}(B^n, \mathbf{R})$ that is almost orthonormal: $< \phi_S, \phi_T >$ is zero if $S \neq T$ and 2^n otherwise. The Fourier transform of a function is just its coordinates in this basis. If we identify the subsets of $\{1, ..., n\}$ with their characteristic Boolean vectors, then the transform itself is in $\mathcal{F}(B^n, \mathbf{R})$.

Definition 5 *For $f \in \mathcal{F}(B^n, \mathbf{R})$, define its Fourier transform $Ff \in \mathcal{F}(B^n, \mathbf{R})$ as*

$$\sum_S < f, \phi_S > A_S$$

2.2. The Eigenspaces of the Fourier Transform

2.2.1. Preliminaries

Let us recall some definitions from linear algebra.

Definition 6 *A nonzero vector v is an* eigenvector *of a linear transformation G if $Gv = cv$ for some nonzero c. Such a c is an* eigenvalue *of G. The set of eigenvectors associated with some eigenvalue forms a subspace.*

Since the Fourier transform F is easily shown to be a linear transformation on $\mathcal{F}(B^n, \mathbf{R})$, it is natural to ask what its eigenvalues and eigenvectors are. We can directly observe that for any function f, $F^2 f = 2^n f$. So if a real c has the property that some nonzero f satisfies $Ff = cf$ then $c^2 = 2^n$. Hence the only possible eigenvalues for f are $\pm 2^{\frac{n}{2}}$.

Definition 7 *When n is fixed, define $c = 2^{\frac{n}{2}}$.*

2.2.2. Tensor Product Decomposition

The function space $\mathcal{F}(B^n, \mathbf{R})$ decomposes nicely as the tensor product (Section 28 in Curtis [5]) of smaller spaces. We can quickly deduce the following from basic definitions.

Lemma 8 *If positive integers $n_1, ..., n_k$ satisfy $n = \sum n_i$, then*

$$\mathcal{F}(B^n, \mathbf{R}) = \mathcal{F}(B^{n_1}, \mathbf{R}) \otimes \mathcal{F}(B^{n_2}, \mathbf{R}) \otimes ... \otimes \mathcal{F}(B^{n_k}, \mathbf{R})$$

Lemma 9 *If $f, g \in \mathcal{F}(B^k, \mathbf{R})$ and $f', g' \in \mathcal{F}(B^{k'}, \mathbf{R})$ then $< f \otimes f', g \otimes g' > = < f, g >< f', g' >$.*

The Fourier transform decomposes similarly.

Definition 10 *Let F_k denote the Fourier transformation on $\mathcal{F}(B^k, \mathbf{R})$*

A straightfoward manipulation of definitions establishes the following.

Theorem 11 *If positive integers $n_1, ..., n_k$ satisfy $n = \sum n_i$ then $F_n = F_{n_1} \otimes F_{n_2} \otimes ... \otimes F_{n_l}$.*

2.2.3. The Eigenspaces

The results in Section 2.2.2 provide a simple way to completely characterize the eigenvectors of F.

Define $P, N \in \mathcal{F}(B^1, \mathbf{R})$ (in the truth table basis) by $N = \begin{pmatrix} 1 - \sqrt{2} \\ 1 \end{pmatrix}, P = \begin{pmatrix} 1 + \sqrt{2} \\ 1 \end{pmatrix}$.

Observe that $< P, N >= 0$ and that $FP = cP$ and $FN = -cN$. Apply Lemma 9.

Lemma 12 *The vectors of the form $G_1 \otimes ... \otimes G_n$, where each $G_i \in \{P, N\}$, form an orthogonal basis for $\mathcal{F}(B^n, \mathbf{R})$.*

Theorem 13 *The eigenvalues of F are $\pm c$. The subspaces of eiegenvectors associated with each have dimension 2^{n-1}.*

Proof: Consider the vectors from Lemma 12. The 2^{n-1} of them with an even number of N factors form an orthogonal basis for the c subspace; the other half form one for the $-c$ subspace. □

Definition 14 *Let us call the subspace of eigenvectors associated with c the* positive eiegenspace *of F. Define the* negative eigenspace *similarly.*

Corollary 15 *Any $f \in \mathcal{F}(B^n, \mathbf{R})$ can be written uniquely as $f^+ + f^-$, where f^+ is in the positive eigenspace and f^- in the negative. Further, $Ff = cf^+ - cf^-$.*

2.3. Boolean Fixed Points

The eigenspaces of F contain Boolean functions that are exactly as computationally hard as we desire. We will demonstrate this by embedding arbitrary Boolean functions in computationally equivalent Boolean fixed points.

In the sequel we will be constructing functions on $n+2$ bits from functions on n bits. For $h \in \mathcal{F}(B^{n+2}, \mathbf{R})$, if the restrictions of h on the last two bits are $h_{--}, h_{-+}, h_{+-}, h_{++}$ then we may write h as the matrix $\begin{pmatrix} h_{-+} & h_{++} \\ h_{--} & h_{+-} \end{pmatrix}$.

Lemma 16 *For $f, g \in \mathcal{F}(B^n, \mathbf{R})$ define $h \in \mathcal{F}(B^{n+2}, \mathbf{R})$ by $h = \begin{pmatrix} f & g \\ g & -f \end{pmatrix}$. Then*

$$Fh = \begin{pmatrix} -2Ff & 2Fg \\ 2Fg & 2Ff \end{pmatrix}$$

Lemma 17 *Let f, g, h be as in Lemma 16. If $Ff = -cf$ and $Fg = cg$ then h is in the positive eigenspace of $\mathcal{F}(B^{n+2}, \mathbf{R})$. If $Ff = cf$ and $Fg = -cg$ then h is in the negative eigenspace.*

Definition 18 *For arbitrary Boolean functions f on n bits, define Boolean functions $FP^+(f)$ and $FP^-(f)$ on $2n+2$ bits inductively as follows. If $n = 0$, f must be constant. Define $FP^+(f) = f\ AND$ and $FP^-(f) = f\ OR$ where AND and OR are functions on 2 bits. If $n > 0$, restrict f on x_n and to obtain $n-1$-bit functions. Then define*

$$FP^+(f) = \begin{pmatrix} FP^-(f|_{x_n=-1}) & FP^+(f|_{x_n=1}) \\ FP^+(f|_{x_n=1}) & -FP^-(f|_{x_n=-1}) \end{pmatrix}$$

$$FP^-(f) = \begin{pmatrix} FP^+(f|_{x_n=-1}) & FP^-(f|_{x_n=1}) \\ FP^-(f|_{x_n=1}) & -FP^+(f|_{x_n=-1}) \end{pmatrix}$$

Lemma 19 *For any Boolean f, $FP^+(f)$ is in the positive eigenspace and $FP^-(f)$ is in the negative.*

Proof: $AND \in \mathcal{F}(B^2, \mathbf{R})$ is in the positive eigenspace and OR is in the negative. Apply Lemma 17. □

Lemma 20 *For arbitrary Boolean f, $f(x_1, ..., x_n) = FP^{\pm}(f)(1, 1, x_1, 1, x_2, 1, ..., x_n, 1)$.*

Proof: Unravel the definition of $FP^{\pm}$: the $x_n, 1$ assigned to the rightmost variables selects the FP of the appropriate restriction of f on x_n. □

It is not surprising that f reduces to its fixed points. What is surprising is that if we can do parity, the fixed points reduce to f.

Lemma 21 *There exist constants c_1, c_2, c_3 such that for any Boolean $f \in \mathcal{F}(B^n, \mathbf{R})$ if*

- *f is computed by a circuit C with g_f gates and depth d_f*
- *parity on n bits is computed by a circuit with g_p gates and depth d_p*

then $FP^+(f)$ is computed by a circuit with not more than $g_f + 2g_p + c_1 n + c_2$ gates and of depth not more than $\max(d_p, d_f) + c_3$. A similar result holds for $FP^-(f)$.

Proof: Label the variables in $\mathcal{F}(B^{2n+2}, \mathbf{R})$ as $< w_1, w_2, y_1, z_1, y_2, z_2, ..., y_n, z_n >$ and use the standard labeling $< x_1, ..., x_n >$ for the variables in $\mathcal{F}(B^n, \mathbf{R})$.

From its definition we see that $FP^+(f)$ equals $\pm FP^{\pm}(f|_{x_n = \overline{y_n \oplus z_n}})$. The sign out front follows $\bar{y_n} \vee z_n$; the sign in the exponent follows $y_n \oplus z_n$. Telescoping, we see that

$$FP^+(f)(1, 1, y_1, z_1, ..., y_n, z_n)$$

can be computed by calculating $f(\overline{y_1 \oplus z_1}, ..., \overline{y_n \oplus z_n})$ and negating this value if there are an odd number of -1s in the n bits $\bar{y_i} \vee z_i$. However, when the first two bits w_1 and w_2 are not both 1, we need to know which 0-bit fixed point was used, *AND* or *OR*. This is determined by the parity of the n bits $y_i \oplus z_i$.

So we calculate $f(\overline{y_1 \oplus z_1}, ..., \overline{y_n \oplus z_n})$, negate depending upon the parity of the $\bar{y_i} \vee z_i$, and then *NOT-EXCLUSIVE-OR* against $AND(w_1, w_2)$ or $OR(w_1, w_2)$ depending on the parity of the $y_i \oplus z_i$. □

Theorem 22 *For any Boolean function f on n bits there exist Boolean function g, h on $2n+2$ bits such that $Fg = cg, Fh = -ch$, f is a polynomial projection of g and h, and g and h both reduce to calculating one instance of f, two instances of parity, and a linear number of two-bit Boolean functions.*

We have also proven [16] a companion result to Theorem 22: representatives of each class exist which cannot be obtained by taking the *Sign* of anything in the eigenspaces.

Theorem 22 gives us the following directly.

Theorem 23 *Let C be a class of functions containing the two-bit functions closed under constant depth, polysize composition. If C contains parity, then $TT(C) \subseteq F(C)^*$.*

In Section 3 we will prove a stronger version of this theorem.

3. A General Theorem

We now generalize the results of Section 2. In particular, we generalize Theorems 22 and 23 to apply for more general families of transforms.

3.1. Preliminaries

3.1.1. Functions

Consider complex functions on some finite group D.

Definition 24 ($\mathcal{F}$) *Let $\mathcal{F}(A,B)$ be the set of functions from A to B.*

Definition 25 (δ) *For $d \in D$ define $\delta_d \in \mathcal{F}(D,\mathbb{C})$ by*

$$\delta_d(x) = \begin{cases} 1 & \text{if } x = d \\ 0 & \text{otherwise} \end{cases}$$

Then $\mathcal{F}(D,\mathbb{C})$ can be conceived of as a $|D|$-dimensional vector space over $\mathbb{C}$ with canonical basis $\{\delta_d : d \in D\}$. The coordinates of a function in this basis is just its truth table: the values it takes on for each element of D.

Further, the standard inner product for complex vector spaces conveniently relates to the correlation between two functions.

Definition 26 *For $f,g \in \mathcal{F}(D,\mathbb{C})$ define their inner product*

$$< f|g > = \sum_{d \in D} f(d)\overline{g(d)}$$

The following is no surprise.

Lemma 27 *The δ basis is orthonormal.*

In our parity work, we had two levels of functions on Boolean variables: those that mapped into the reals, and those that mapped into ± 1—the natural embedding of the Booleans in the reals. The surprise came in that while transforms of the smaller level are generally in the larger, there existed functions in the small level whose transforms (when multiplied by a constant) were still in the small level.

This suggests that we want to identify a special subclass of functions in $\mathcal{F}(D,\mathbb{C})$.

Definition 28 (W, W', k) *Let W be a finite group, and let W' be a representation of it in $\mathbb{C}$. Let $|W| = k$.*

Because W is a finite group, we have that $w \in W' \Rightarrow \bar{w} \in W'$. Also, $0 \in W$ maps to $1 \in W' \subseteq \mathbb{C}$.

So our special class will be $\mathcal{F}(D,W')$. In general, D will turn out to be n-tuples of W (that is, we currently cannot think of anything else interesting for D to be).

3.1.2. Transforms

We need to generalize the idea of the parity transform.

In the parity case, we specified an alternate orthonormal basis for the general function class (i.e., $\mathcal{F}(D,\mathbb{C})$). The functions in the new basis were all in the smaller class (i.e., $\mathcal{F}(D,W')$).

So we need to specify a new basis for $\mathcal{F}(D,\mathbb{C})$ consisting of functions from $\mathcal{F}(D,W')$. In order for the transform to be a self-map of the function space, we need to identify each new basis function with an element of D. Since there are $|D|$ elements in the new basis, we specify this identification explicitly by indexing the new basis by D— $\{\phi_d : d \in D\}$.

The complexity fixed point theorem for parity works by considering complexity classes that contain parity. Conveniently, the basis functions for the parity transform are merely smaller versions of parity. When we generalize the transform this property may no longer hold, so we need a way of generalizing the condition on the complexity class.

Since the new basis consists of $|D|$ functions, indexed by D, from D to W', we can regard the basis as a whole as a function from D^2 to W'.

We can now make the following definitions.

Definition 29 *Let $\phi \in \mathcal{F}(D^2,W')$. Let ϕ_d be the restriction of ϕ on its first variable—e.g., $\phi_{d_1}(d_2) = \phi(d_1,d_2)$. The transform F_ϕ induced by ϕ on $\mathcal{F}(D,\mathbb{C})$ is defined by*

$$F_\phi f \;=\; \sum_{d\in D} < f|\phi_d > \delta_d$$

The transform is orthonormal when the ϕ_d satisfy

$$< \phi_{d_1}, \phi_{d_2} >= \begin{cases} |D| & \text{if } d_1 = d_2 \\ 0 & \text{otherwise} \end{cases}$$

We will write F for F_ϕ when ϕ is understood.

3.1.3. Bigger Domains

We observe that, as in all complexity theory, we are only pretending to be working with one particular $\mathcal{F}(D,W')$. In reality, we have a family $\mathcal{D}$ of domains of increasing size. The domains—usually tuples of variables—are built up by adjoining smaller domains. Functions on the smaller domains can be combined to obtained functions on the larger in a natural way using tensor products.

Similarly, we talk about a function ϕ that operates on $D \times D$ but we really mean a family of ϕ, one for each D. The orthogonal transform ϕ on a family of domains $\mathcal{D}$ acutally consists of a family of transforms, one for each $D \in \mathcal{D}$.

There is a natural way to combine orthogonal transforms on two domains D_1 and D_2 to obtain one on $D_1 \times D_2$. We engage in a slight abuse of notation.

Definition 30 *Suppose ϕ_1 and ϕ_2 are transforms on $\mathcal{F}(D_1, \mathbb{C})$ and $\mathcal{F}(D_2, \mathbb{C})$. We can define a transform $\phi_1 \otimes \phi_2$ on the function space $\mathcal{F}(D_1 \times D_2, \mathbb{C})$ by*

$$(\phi_1 \otimes \phi_2)(< a_1, a_2 >, < b_1, b_2 >) \;=\; \phi_1(a_1, b_1)\phi_2(a_2, b_2)$$

for $a_i, b_i \in D_i$.

We used tensor product notation for the combined transform because in pretty much every sense, the transform is a tensor product. Each basis element in the new basis is the tensor product of basis elements in the smaller ones. Further, there is the following directly obtained fact.

Lemma 31 *For $< f_1, f_2 > \in \mathcal{F}(D_1 \times D_2, \mathbb{C})$,*

$$F_{\phi_1 \otimes \phi_2} < f_1, f_2 > \;=\; F_{\phi_1} f_1 \;\otimes\; F_{\phi_2} f_2$$

Orthogonality and orthonormality are also preserved.

Lemma 32 *If ϕ_1 and ϕ_2 are orthogonal [orthornormal] transforms, then so is $\phi_1 \otimes \phi_2$.*

So when we talk about about a transform ϕ, we really mean a hierarchy of transforms, one for each domain.

Definition 33 *Suppose for all domains D, the transform ϕ on D^2 is just the tensor square of the transform ϕ on D, as per Definition 30. We then say that ϕ is* multiplicative.

3.2. The General Theorem

In this section we prove our general fixed-point theorem. We actually obtained this result by directly generalizing the Boolean parity case. Once we obtained this result, it became possible to prove it directly. We use the direct approach here.

Theorem 34 *Let $f \in \mathcal{F}(D, W')$ and let F_ϕ be an orthogonal transform. Then there exist $g, h \in \mathcal{F}(D^2, W')$ such that*

- *g, h reduce to each other and ϕ*
- *$F_{\phi \otimes \phi} g = |D| h$*

- *f is a polynomial projection of g*
- *g and h reduce to f and ϕ*

If $\phi(a,b) = \phi(b,a)$ then g, h reduce to just each other.

Proof: Given f and ϕ, define

$$g(a,b) = f(a)\phi(a,b), \quad h(a,b) = f(b)\overline{\phi(a,b)}$$

Then

$$\begin{aligned}
(F_{\phi\otimes\phi}g)(a,b) &= < g|\phi\otimes\phi_{a,b} > && \text{(by definition of } F_{\phi\otimes\phi}) \\
&= \textstyle\sum_{c,d\in D}\ g(c,d)\ \overline{\phi\otimes\phi(\ (a,b),\ (c,d)\ )} && \text{(by definition of inner product)} \\
&= \textstyle\sum_{c,d\in D}\ f(c)\phi(c,d)\ \overline{\phi\otimes\phi(\ (a,b),\ (c,d)\ )} && \text{(by definition of } g) \\
&= \textstyle\sum_{c,d\in D}\ f(c)\phi(c,d)\ \overline{\phi(a,c)}\ \overline{\phi(b,d)} && \text{(by definition of } \phi\otimes\phi) \\
&= \textstyle\sum_{c\in D}\ f(c)\ \overline{\phi(a,c)}\ \sum_{d\in D}\ \phi(c,d)\ \overline{\phi(b,d)} && \text{(rearranging sums)} \\
&= \textstyle\sum_{c\in D}\ f(c)\ \overline{\phi(a,c)}\ < \phi_c|\phi_b > && \text{(by definition of inner product)} \\
&= |D|\ f(b)\ \overline{\phi(a,b)} && \text{(by orthonormality of } \phi)
\end{aligned}$$

□

We can use the existence of g and h to demonstrate the existence of complexity classes that are fixed under the truth table to spectral transformation.

Theorem 35 *Let W be a finite group and let ϕ be a multiplicative orthonormal transformation on $\mathcal{F}(D,W')$. Suppose $C \subseteq \mathcal{F}(D,W')$ satisfies*

- *$C = C^*$*
- *C can calculate W-negation and 2-way W-addition.*

If C can calculate ϕ, then

$$TT(C) \ \subseteq \ F_\phi(C)^*$$

3.3. Group Characters

Now that we have established a nice theorem about orthogonal transforms and $\mathcal{F}(D,W')$ for finite groups W, it would be nice to demonstrate the existence of interesting transforms that meet the criteria of the theorem.

Group representation theory [12] [13] provides such a family of tranforms. The simple characters of a group W are complex functions on W that are pairwise orthogonal. The

inner product of a simple character with itself is k. Suppose $W_1, ..., W_r$ are the conjugacy classes of W: then there are r simple characters, each of which is fixed on each W_i. Further, simple characters of W combine to form simple characters on W^n exactly the way we want them to (eg, as per Definition 30): they multiply.

This suggests that if the simple characters are to form the alternate basis for our function space $\mathcal{F}(W^n, W')$, then the original basis better have dimension r^n and the functions better be fixed on each conjugacy class. For all practical purposes, the variables range over the conjugacy classes of W rather than the individual elements. Having each element be its own conjugacy class would make life much nicer; this happens exactly when W is an abelian group. A result from algebra tells us that any finite abelian group isomorphic to the direct product of cyclic groups. A standard result from group representation theory (eg, Theorem 2.4 in Ledermann [12]) tells us exactly what the simple characters of W are.

Lemma 36 *Suppose*

$$W = \mathbb{Z}/k_1 \times \mathbb{Z}/k_2 \times ... \times \mathbb{Z}/k_s$$

Then the simple characters of W are the $\{\chi_w : w \in W\}$, where

$$\chi_w(x) = e^{2\pi i \sum_{j=1}^{s} \frac{x_j w_j}{k_j}}$$

So we can define $\phi(a, b)$ to be $\chi_a(b)$ and obtain an orthogonal transform for $\mathcal{F}(W, W')$ and hence for $\mathcal{F}(W^n, W')$.

Corollary 37 *Suppose W is a finite abelian group and ϕ is the orthogonal transform of $\mathcal{F}(W^n, W')$ induced by the simple characters of W. Theorem 34 and Theorem 35 hold for W and ϕ.*

The fixed point property for the Boolean parity transform is just Corollary 37 with $W = \mathbb{Z}/2$.

4. Further Work

This work suggests a couple of unanswered questions.

First, for completeness' sake, we wonder whether the converse to Theorem 35 is true: that ϕ is the easiest fixed point.

Conjecture 38 *Let C be a complexity class and ϕ a sufficiently nice orthogonal transformation. Then if there exist $g, h \in C$ with $F_\phi g = ch$ then $\phi \in C$.*

In the Boolean parity case, the parity spectra had fixed points for classes that contained parity but yielded meaningful invariants for classes that didn't—in particular, for $AC0$. Proving this conjecture would provide theoretical evidence that the ϕ spectra will yield similar invariants for more general classes. What these clases are, and whether the invariants tell us anything interesting, is another question altogether, and one which we will explore shortly.

The second avenue for further inquiry is generalizing the family of ϕ to which our theorem applies. The multiplicative property is rather limiting: we never move beyond multiplication in $\mathbb{C}$. However, in principle we only need some kind of self-similarity between ϕ on larger domains and ϕ on smaller ones. The relation does not really have to be multiplicative.

5. References

1. Ahmed, Nasir and Kamisetty R. Rao. *Orthogonal Transforms for Digital Signal Processing.* Berlin: Springer-Verlag, 1975.

2. Boerner, Hermann. *Representations of Groups.* Amsterdam: North-Holland, 1963.

3. Bollobas, Bela. *Combinatorics.* Cambridge: Cambridge University Press, 1986.

4. Bruck, Jehoshua and Roman Smolensky. "Polynomial Threshold Functions, $AC0$ Functions, and Spectral Norms." 31st FOCS, 1990.

5. Curtis, Charles W. *Linear Algebra: An Introductory Approach.* New York: Springer-Verlag UTM, 1984.

6. Furst, Merrick, Jeff Jackson and Sean Smith. "Improved Learning of $AC0$ Functions." COLT, 1991.

7. Gotsman, Craig. "On Boolean Functions, Polynomials, and Algebraic Threshold Functions." Jerusalem: Institute of Mathematics and Computer Science, the Hebrew University. (rough draft)

8. Gotsman, Craig and Nathan Linial. "Spectral Properties of Threshold Functions." Submitted to *Combinatorica*, 1990.

9. Hall, Marshall. *Combinatorial Theory.* Waltham, MA: Blaisdell Publishing, 1967.

10. Hoffmann, Kenneth and Ray Kunze. *Linear Algebra.* Englewood Cliffs: Prentice-Hall, 1971.

11. Kahn, Jeff, Gil Kalai and Nathan Linial. "The Influence of Variables on Boolean Functions." 29th FOCS, 1988.

12. Ledermann, Walter. *Introduction to Group Characters.* Cambridge: Cambridge University Press, 1977.

13. Leech, J.W. and D.J. Newman. *How to use Groups.* London: Methuen, 1969.

14. Linial, Nathan, Yishay Mansour and Noam Nisan. "Constant Depth Circuits, Fourier Transform, and Learnability." 30th FOCS, 1989.

15. Miller, Willard. *Symmetry Groups and their Applications.* New York: Academic Press, 1972.

16. Smith, Sean and Carl Sturtivant. "Fixed Points under the Fourier Transform of Boolean Functions." Pittsburgh: CMU-CS-90-192 (technical report).